

Amendments to the Claims

This listing of claims will replace all prior versions, and listings, of claims in the application:

Listing of Claims:

Claims 1 and 2: (Canceled).

- 1 3. (Currently amended) The method of ~~claim 2~~claim 23, wherein said forming said
2 ordered set of deadlock-free sub-topologies of said network further comprises
3 forming at least one sub-topology of said network that is a spanning layer of said
4 network.
- 1 4. (Original) The method of claim 3, further comprising maintaining a cost of a
2 corresponding link between each of said nodes in said network within each entry
3 of said routing table.
- 1 5. (Original) The method of claim 4, wherein said adding said column to said routing
2 table further comprises:
3 for each entry within said column, performing the following steps
4 determining a set of cost values, wherein each value within said set
5 of cost values reflects a sum of the cost of reaching a selected neighbor
6 node of said node from said corresponding forwarding node and the cost
7 of reaching said node from said selected neighbor node,
8 determining a minimum value of said set of cost values, and
9 determining forwarding information for said entry indicating said
10 selected neighbor node associated with said minimum value.
- 1 6. (Original) The method of claim 4, wherein said adding said row to said routing
2 table further comprises:
3 for each entry within said row, performing the following steps

4 determining a set of cost values, wherein each value within said set
5 of cost values reflects a sum of the cost of reaching said corresponding
6 destination node from a selected neighbor node of said node and the cost
7 of reaching said selected neighbor node from said node,
8 determining a minimum value of said set of cost values, and
9 determining forwarding information for said entry indicating said
10 selected neighbor node associated with said minimum value.

Claims 7 and 8: Canceled

1 9. (Currently amended) The system of ~~claim 8~~claim 21, wherein said routing logic is
2 further operable to form said ordered set of deadlock-free sub-topologies of said
3 network further by forming at least one sub-topology of said network that is a
4 spanning layer of said network.

1 10. (Original) The system of claim 9, wherein said routing logic is further operable to
2 maintain a cost of a corresponding link between each of said nodes in said
3 network within each entry of said routing table.

1 11. (Original) The system of claim 10, wherein routing logic operable to add said
2 column to said routing table is further operable to perform the following steps for
3 each entry within said column:

4 determine a set of cost values, wherein each value within said set of cost
5 values reflects a sum of the cost of reaching a selected neighbor node of said
6 node from said corresponding forwarding node and the cost of reaching said
7 node from said selected neighbor node;

8 determine a minimum value of said set of cost values; and

9 determine forwarding information for said entry indicating said selected
10 neighbor node associated with said minimum value.

1 12. (Original) The system of claim 10, wherein said routing logic operable to add said
2 row to said routing table is further operable to perform the following steps for
3 each entry within said row:

4 determine a set of cost values, wherein each value within said set of cost
5 values reflects a sum of the cost of reaching a corresponding node from a
6 selected neighbor node of said node and the cost of reaching said selected
7 neighbor node from said node;

8 determine a minimum value of said set of cost values; and

9 determine forwarding information for said entry indicating said selected
10 neighbor node associated with said minimum value.

1 13. (Previously presented) The system of claim 24, wherein said routing logic
2 comprises at least one digital logic circuit.

1 14. (Previously presented) The system of claim 24, wherein said routing logic
2 comprises program code loaded into a memory of a computer system.

1

2 Claims 15 – 20: Canceled

1 21. (Previously presented) The method of claim 23, further comprising iteratively
2 performing said steps of adding a row of entries and adding a column of entries
3 in order to add routing information to said routing table for a plurality of nodes.

1 22. (Previously presented) The method of claim 23, wherein said existing deadlock-
2 free set of paths are through a network of two nodes.

1 23. (Currently amended) A method for adding routing information for a new node to a
2 routing table with a plurality of entries that reflect an existing deadlock-free set of
3 paths through a network of nodes, wherein the routing table has a row for each
4 source node in the network and a column for each destination node in the
5 network and wherein a table entry located at an entry row and an entry column

identifies a link that can be used to send data from the source node in the entry row to the destination node in the entry column, the method comprising:

forming an ordered set of deadlock-free sub-topologies of said network,
each sub-topology comprising links that are not used in any other sub-topology;
generating said routing table as a function of said ordered set of deadlock-free sub-topologies;
adding to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and
adding to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node,
wherein the paths defined in the routing table continue to define deadlock-free paths in the network after addition of the row and column for the new node.

24. (Currently amended) A system for adding routing information for a new node to a routing table with a plurality of entries that reflect an existing deadlock-free set of paths through a network of nodes, wherein the routing table has a row for each source node in the network and a column for each destination node in the network and wherein a table entry located at an entry row and an entry column identifies a link that can be used to send data from the source node in the entry row to the destination node in the entry column, comprising routing logic operable to:
- form an ordered set of deadlock-free sub-topologies of said network, each sub-topology comprising links that are not used in any other sub-topology;
generate said routing table as a function of said ordered set of deadlock-free sub-topologies;
add to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can

be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and

add to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node,

wherein the paths defined in the routing table continue to define deadlock-free paths in the network after addition of the row and column for the new node.

25. (Currently amended) A system for adding routing information for a new node to a routing table with a plurality of entries that reflect an existing deadlock-free set of paths through a network of nodes, wherein the routing table has a row for each source node in the network and a column for each destination node in the network and wherein a table entry located at an entry row and an entry column identifies a link that can be used to send data from the source node in the entry row to the destination node in the entry column, comprising:

means for forming an ordered set of deadlock-free sub-topologies of said network, each sub-topology comprising links that are not used in any other sub-topology;

means for generating said routing table as a function of said ordered set of deadlock-free sub-topologies;

means for adding to the routing table, a row including a plurality of entries, each entry identifying a link that directly connects the new node to a neighbor node that can be connected, via existing deadlock-free paths described by the table, to a destination node associated with the entry column; and

means for adding to the routing table a column including a plurality of entries, each entry identifying a link that can be used to connect a source node associated with the entry row, via existing deadlock-free paths described by the table, to a neighbor node that can be directly connected to the new node,

wherein the paths defined in the routing table continue to define deadlock-free paths in the network after addition of the row and column for the new node.

1 26. (Currently amended) A computer program product including a computer readable
2 medium, said computer readable medium having a computer program stored
3 thereon, said computer program for adding routing information for a node to a
4 routing table, wherein said routing table includes routing information reflecting an
5 existing deadlock-free set of paths through a network of nodes, said computer
6 program comprising:

7 program code for forming an ordered set of deadlock-free sub-topologies
8 of said network, each sub-topology comprising links that are not used in any
9 other sub-topology;

10 program code for generating said routing table as a function of said
11 ordered set of deadlock-free sub-topologies;

12 program code for adding to the routing table, a row including a plurality of
13 entries, each entry identifying a link that directly connects the new node to a
14 neighbor node that can be connected, via existing deadlock-free paths described
15 by the table, to a destination node associated with the entry column; and

16 program code for adding to the routing table a column including a plurality
17 of entries, each entry identifying a link that can be used to connect a source node
18 associated with the entry row, via existing deadlock-free paths described by the
19 table, to a neighbor node that can be directly connected to the new node,

20 wherein the paths defined in the routing table continue to define deadlock-
21 free paths in the network after addition of the row and column for the new node.

Claim 27: Canceled